Exercise Set 7 - Liveness and Asynchrony

Exercise 1

Answer the following questions:

- When should you consider using asynchronous invocations?
- In what sense can a direct invocation be asynchronous?
- What is an “early reply”?  
- What are “futures”?
- When are futures better than early replies?

Exercise 2

The situation is critical, ghosts are invading the city. You and four friends have been able to hide in an abandoned house.

The house has four entries. You have a single ghostbusters weapon so you have developed a plan. Each of your four friends controls one of the four entries; they let in individual ghosts, keeping count of how many have entered. You stand in the center, and eliminate the ghosts that have entered as fast as you can, keeping track of how many you have removed. You don’t want too many ghosts in the room for obvious reasons, and so must periodically check on how many ghosts are in the building in total. If there are too many you need to ensure no new ghosts enter until you’ve had opportunity to reduce their numbers. For this you can only radio each of your friends individually to find out how many they have let in and/or ask them to close the entry. Only once the total number is below a reasonable threshold should you allow more ghosts back in, again only by radioing each friend individually.

Simulate this as a multi-threaded program. You should have 5 threads, one representing you and one for each friend/door. Each friend thread should let in a ghost with a 10% probability every 10ms, keeping track of the number she admitted. The thread representing you has a 40% probability of removing a ghost once every 10ms. You should check the total every 2s, and while waiting for the response continue to eliminate the ghosts. When you get the information about the total number of ghosts, if the number is below n then everything is ok, otherwise no new ghosts should be admitted until you’ve reduced the number to below n/2. n is a command-line parameter.